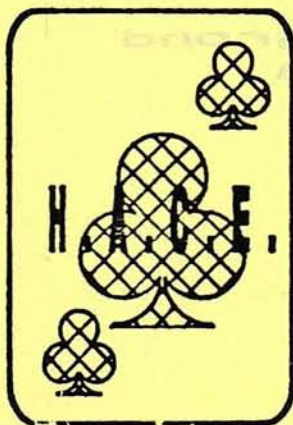


HOUSTON ATARI COMPUTER ENTHUSIASTS



AUGUST 1987

NEWSLETTER

Houston ATARI Computer Enthusiasts
P.O. Box 562
Katy, Tx 77492-0562

DATED MATERIAL - DELIVER PROMPTLY

BULK RATE
U.S. POSTAGE
PAID
Houston, Tx.
Permit No. 6257

.....newsletter exchange.....

S.L.C.C.
P.O. BOX 1525
SAN LEANDRO, CA

94577

CALL EITHER OF THE TWO HACE BBS SERVICES
H.A.C.E. BBS - 713-583-8980 : HACE-ST BBS - 713-827-8041

You will notice that HACE has moved its monthly meeting place. During 1987 we will meet at Stouffer's Greenway Plaza Hotel - Normally we will have the Bluebonnet Room which is located in the lower level. There will be NO CASH BAR since it was the consensus of most members that this actually can be a deterrent. Complimentary parking will be provided by the Hotel in its underground parking area.

AUGUST MEETING

6:45 P.M.

Wednesday
August 23, 1987

Bluebonnet Room
Stouffers Hotel

Greenway Plaza
Edloe @ Southwest Freeway

Free Parking in Undergrond
Hotel Parking Area

Editors Corner

*HOW TO MAIL 150 NEWSLETTERS

1. Sort by ZIP code and take to Bulkmail at Central Post Office at nine in the morning.
2. Be rejected at bulkmail 'cause the minimum is 200 pieces.
3. Get one newsletter weighed to find out correct postage.....Wow! just under one ounce, only 22 cents needed.
4. Rush back to Alief, get page of unused 39 cent stamps left over from previous mailing.
5. Exchange 39 cent stamps for 22 cent stamps.
6. Handstamp each newsletter being careful to cover up bulk mail notice.
7. Put newsletters back in ZIP code order after fat lady accidentally pushes them off postoffice counter.
8. Give 'em to postal clerk who says "No sweat, it's still before noon, they will be sorted and out today."
9. Four days later, get call from Club President....Postal Inspectors have intercepted newsletters and he had to put an extra dime stamp on each 'cause the format of the title page is vertical not horizontal
10. Listen to members wonder why it takes ten days to get newsletters.

DBASIC --- A REVIEW OF "8K-BASIC" FOR THE ST!

Really, that's meant as a complement. When I first booted up the New Atari's kludge of DRI's BASIC, my first thought was "Oh God!"; and the more I used it, the more I hated it and longed for something as simple as the old 8-K BASIC. Simplicity, and the good points of the old BASIC, with most of the warts ground down, have re-appeared in the form of DBASIC from an outfit called DTACK Grounded Inc.

First, I'll begin with a disclaimer and an excuse. I was given the 280-page manual, a disk, and instructed to produce a quick review for the newsletter. This article is based on a perusal of the manual and some, but not much, actual experience with the language. Only the high spots will be touched and any statements made herein are subject to radical change as experience is gained. So much for the weasel words.

DBASIC is both a language and a different DOS. Gone is GEM. When DBASIC is booted the mouse can hide in its hole, for it is useless in the DBASIC environment. Gone also are windows -- especially that da___d, clumsy EDIT window. Full-screen editing is back! (Cheers all round). A sort of "twister" formatting scheme is used, which means the program is fast booting. Disk sector re-allocation is eliminated. This means that each "save" of your program involves writing a whole new copy of the program to disk -- without reclaiming the sectors used by the older version. For the usual simple BASIC programs this shouldn't be too much of a burden, there's a neat utility to quickly copy only active files from one disk to a blank disk. However, if you are changing and storing huge data arrays in your program, you will find that the disk fills rapidly. DTACK claims, rightly, that read/write speed is gained by not having to consult sector link tables and move the head all over the disk. I'm also sure that it is much easier to write a DOS without having to worry about sector links (we all know what "features" are).

DBASIC is written in assembly language, which means it is compact and fast. What we have here is an interpreted BASIC which compiles each line to p-code AS IT IS ENTERED! LISTing a line involves disassembling p-code on the fly. This is really quite a feat. You get the speed of a

compiled BASIC without the hassle of the "compile" step. Actually DTACK claims, with benchmark result support, that DBASIC rivals compiled "C" code in speed, and easily outperforms other BASICs, especially GFA compiled code, as well as FAST-BASIC. For my simple needs, I've thought that ST-BASIC was fairly fast, but I'll admit to wishing for more speed in 8K BASIC. But DBASIC is impressive. A simple real-life example: My programs contain instructions and other reading material. Often I use a "do nothing" loop to hold the screen for reading before continuing the program. In 8K BASIC a 4-500 cycle loop was usually sufficient. ST-BASIC needs about a 4000 cycle loop, while with DBASIC, a 32000 cycle loop is still too short. Line numbers are used. I like using line numbers in BASIC, which is why I haven't bought GFA BASIC

Both integer (FAST) and floating point (less FAST) arithmetic are supported. But the two cannot be mixed. There are commands to convert one to the other to allow arithmetic to be done. Floating point arithmetic uses 64-bit (double precision) representations of a number in all operations. Output display however is rounded to 12 significant digits. The usual functions and operations are allowed.

Variable types must be individually declared -- no global declarations as in Microsoft Basic.

Arrays in DBASIC are neat. How about 15-dimension string arrays? I admit to being unable to visualize that -- let alone keep track of them. Array size is unlimited, except of course for machine memory limitations (compare that to the 32K limit for ALL arrays in ST-BASIC). String length is limited to 255 characters. No 30,000 character strings a la 8K BASIC--darn. But multi-dimensional string arrays go a long way towards making up for that. Default string length is 255 characters, but that length can be globally or individually reduced.

There is no PRINT USING command, but then the version in ST-BASIC is no great deal. Besides, we 8K programmers know how to get around that lack. See, skills can be transferred. Also missing are OR and AND commands for use with IF....THEN -- but

there is an ELSE command available. RIGHT\$, LEFT\$, and MID\$ commands are allowed. Cursor positioning for text is performed by "Apple-ish" HTAB and VTAB. There is a command to erase a text line. PEEKS and POKES are supported, both with bytes and words. There is a command to view the disk catalog without exiting BASIC, but none to format a new data disk -- as was learned when I discovered that the program disk was full upon trying to save a program.

Since there is no TOS or GEM available in the DBASIC environment, there are no windows, dialog boxes, or polylines possible. But have you ever managed to use them from within ST-BASIC? There are DBASIC commands to position and plot lines, rectangles, circles, and pixels; as well as others to erase what was drawn. For graphics, color can be specified and changed by COLOR and PALETTE commands. No FILL-type commands though. I have not been able to change the text color so far and suspect that ability was left out. But, how about this? Changing from a high to a lo-res color screen is as easy as MODE 0. Try that little trick in ST-BASIC. However, there is no direct support of mixed mode screens.

Unfortunately your ST is rendered mute -- there are no sound commands in DBASIC. There is, however, a hint in the manual of a sound effect using POKES.

How stable and polished is the product? It crashed on me once, locking up the keyboard, as I was trying to convert an 8K program to DBASIC. Why did it crash? I dunno. I don't think I did anything drastically wrong. At that point, I decided that I had better quit playing and begin writing the promised review. So beware -- at least a little.

What you get with ST-BASIC is a disk which can be copied using an included utility and a 280-page manual. The manual is well written with a whimsical twist. Explanations of the language and command usage, as well as the philosophy of DBASIC are generally readable. ANY explanation of hardware registers, including this one, has a severe soporific effect on me. The manual is required reading. In reality DTACK is selling the manual and giving away the disk.

A GOOD DEAL:

This brings us to price. The normal sales

price for the language is \$40. To try to stir interest and win converts, DTACK is giving users groups a disk and twenty copies of the manual to do with as they choose. HACE chooses to sell a disk and the manual for \$10. Omitting my copy, there will be 19 copies of the language for sale at the coming meeting. At that price it's a steal. Forty dollars isn't too bad either. I urge you to at least test drive the language. It seems to be a good environment, with the "look and feel" of 8K BASIC, and it is FAST. Anything is better than ST-BASIC, and even at \$40, DBASIC is one of the cheaper alternatives going. If nothing else, all you secret BASIC lovers can come out of the woodwork and resume programming in a old familiar language. You don't really have to learn "C" to do anything with your ST.

Stephen Novak

CHECK OUT THE LIBRARY

If you aren't picking up the "Disk of the Month", and the monthly magazine disks, you are really missing out on a good deal. The librarian usually has disks that tie to the articles in the newsletter and to subjects discussed at the meetings.

And, to put the truth to it, some public domain and shareware programs like 1030 express, Speedprint, or the Tapeless Spreadsheet Printer beat the @'&' out of commercially available programs

The STs have plenty of software on the market shelves right now, but for the eight bits, libraries like ours are a great resource to make our computer useful and fun.

If you are a narrative game freak, check out the games form the old and sadly defunct Softside's... They might not parse as well as the new stuff, but for original situations and challenges, they are still fascinating.

Eight Bit ARChiving

During the past year I have watched my 800XL become more and more like my 1040 ST. We now have whole track disk copiers, new filetransfer protocols, and archive programs that get the work done in half the time, half of the space, or both.

What makes it especially nice is that it is all done without any hardware upgrades to the 8 bit line and that the necessary software is public domain. Maybe we don't have to rush out and upgrade the 1200 baud modem to 2400 baud, or get a new machine after all.

Although the programs are free, there is a price to be paid. Confusion sometimes results since these programs come with little in the way of documentation or examples. The three productivity programs mentioned earlier are common subjects of misunderstanding on the bulletin boards.

This article will concern itself with the archive process. ARC has been in use in the CP/M and MS-DOS world for many years. It is a library and compression utility in which files are reduced 45-55%. This means that programs can be stored in half the space and transferred in half of the normal time. This becomes especially significant when using the phone lines to transfer information via modems.

In researching this article, I called a local ATARI user group Bulletin Board System to download a 60,000 character file describing the ARChive process. Although originally 30 pages of typewritten text, it was transferred from this BBS to my computer in less than two minutes in compacted form.

Even more important to me was the fact that my 8 bit machine was able to use this ARC'ed text file. Until last year when Ralph Walden published his ARChive program for the Atari, 95% of the information on IBM and ST bulletin boards was off limits to us 8 bitters since they require that programs sent to them be compressed.

ARC differs from programs such as SHRINK, SCOPY, and SCRUNCH in that those programs are for whole disks and don't compress any data. In fact, it is common that a full disk (720 sectors) can end up larger when a single file is made using those other "compacting programs".

ARC and those other programs each have a specific purpose. ARC is for combining and reducing multiple files while SHRINK, SCOPY, and SCRUNCH copy everything on a disk. As beginners, I'm sure that we all once tried copying DOS.SYS and DUP.SYS from one disk to another and discovered that our "duplicate" would not boot up when we turned on the computer.

We quickly learned that there were many things on our "blank disk" that were hidden. Even files that had been deleted were still actually there although they didn't show. The whole disk copy programs get everything that is on the disk whether that's what you want or not. By creating one single file, you can then store an entire disk on a hard drive or send the whole thing to a BBS for others.

ARC, however, will only work on files displayed from the directory. This is especially useful for combining (and then reducing) files that should be together such as a program and its documentation. I couldn't count the number of times that I have received a program that turned out to be unuseable because the accompanying instruction file was missing.

ARC for the XL and XE computers has only one disadvantage right now and that is the time spent decoding the half size file. The time spent converting back to original form almost negates the time saved downloading a file at 1200 baud. This will be corrected when a 130XE version is released.

On the BBSes, the contents of ARC'ed files can be examined as well as tested for being a successful upload. That means no more bad files that won't run. For lucky owners of hard drives, ARC'ed files mean \$400 units to store the same amount of programs as \$1000 devices. And for most of us, it means the end of "1 program, 1 disk".

For the 8 bit Atari owners, I will upload the ARC.COM and the ARCX.COM (extractor) programs to the HACE BBS at 470-2511. Also look for the text file ARCD0C. ST owners will find ARC.TTP and various "shell" programs for convenient use on the 16 bit HACE board at 827-8041.

Any questions should be addressed to Greg McIntyre. Next month, YMODEM modem protocol to cut your long distance phone bill in half. Any additional topics will be considered ... better yet, submit an article of your own for the HACE newsletter. Contact editor Jim Salmon or send your article to either club BBS.

ARC and those other programs each have a specific purpose. ARC is for combining and reducing multiple files while SHRINK, SPCOPY, and SPCRUNCH copy everything on a disk. As beginners, I'm sure that we all once tried copying DOS.SYS and DOS.SYS from one disk to another and discovered that our "duplicate" would not boot up when we turned on the computer.